Enabling Enterprise Semantic Search through Language Technologies: the ProgressIt Experience

Roberto Basili[1], Andrea Ciapetti[2], Danilo Croce[1],
Valeria Marino[3], Paolo Salvatore[3] and Valerio Storch[1]

[1]Department of Enterprise Engineering, University of Roma, Tor Vergata
{basili,croce,storch}@info.uniroma2.it
[2]Innovation Engineering srl, Roma
a.ciapetti@innovationengineering.eu
[3]Ciaotech srl, Roma
{V.Marino,P.Salvatore}@ciaotech.com

Abstract. This paper presents the platform targeted in the PROGRESS-IT project. It represents an Enterprise Semantic Search engine tailored for Small and Medium Sized Enterprises to retrieve information about Projects, Grants, Patents or Scientific Papers. The proposed solution improves the usability and quality of standard search engines through Distributional models of Lexical Semantics. The quality of the Keyword Search has been improved with Query Suggestion, Expansion and Result Re-Ranking. Moreover, the interaction with the system has been specialized for the analysts by defining a set of Dashboards designed to enable richer queries avoiding the complexity of their definition. This paper shows the application of Linguistic Technologies, such as the Structured Semantic Similarity function to measure the relatedness between documents. These are then used in the retrieval process, for example to ask the system for Project Ideas directly using an Organization Description as a query. The resulting system is based on Solr, inheriting its highly reliability, scalability and fault tolerance, providing distributed indexing, replication and load-balanced querying, automated failover and recovery, centralized configuration and more.

1 Introduction

Innovation is an unstructured process in most of Small and Medium Sized Enterprises (SMEs). The so called "Innovation Management Techniques", considered by the European Commission a useful driver to improve competitiveness, are still underutilized by SMEs. Such techniques include Knowledge Management, Market Intelligence, Creativity Development, Innovation Project Management and Business Creation. However, within these techniques, the Creativity Development techniques are the less used among SMEs[1]. The only activity performed by

[1] European Commission, DG Enterprise Innovation management and the knowledge driven economy - January 2004

almost all SMEs is the search for external information in different sources such as the web, patent databases, in trade fairs or discussing with clients and partners. The main source of information for SMEs is the Internet search [3], an activity realized by more than 90% of SMEs when dealing with innovation. Knowledge and information are often distributed in heterogeneous and unstructured sources across networked systems and organizations. Search for entities (such as competitors or new products) is not always sufficient as search for knowledge, as the one related to novel processes or brands and marketing analysis (whereas connected to large scale opinion mining) is based upon richer information.

The system targeted in the PROGRESS-IT project, funded by Regione Lazio (FILAS-CR-2011-1089) is here presented and discussed as a potential support the SMES during the activities described above. It makes use of some of the results of the European INSEARCH EU project[2], presented in [8] whose focus is the design and development of a useful Search Platform for the SMEs. The PROGRESS-IT platform automatically collects documents expressing Project Ideas, Organization descriptions, Grants, Patents, Scientific Papers and Work Programmes. Three domains have been targeted: Aerospace, ICT and Security. The system has been designed to access this huge amount of information through Standard or Advanced Information Retrieval techniques.

This paper discusses the application of Distributional models of Lexical Semantics [9, 15] to improve of the quality of the retrieval process: the *Keyword Search* is extended with an effective query expansion and re-ranking strategy; moreover, the distributional approaches support the design of several *Dashboards* to enable richer queries avoiding the complexity of their definition.

One the one hand Distributional models of Lexical Semantics [9, 15] have been applied in the *Keyword Search* to improve the quality of the ranking function and providing an effective query expansion. The main idea is that many documents are not retrieved by analysts as they are not able to list all possible query terms to express their information need. Distributional models are used in the system to retrieve additional terms that are paradigmatically similar to the query, e.g., quasi-synonym as discussed in [15]. The precision drop has been balanced by introducing a re-ranking function: the query is projected in the Lexical Semantic Space, as well as the retrieved documents, and their similarity is used to prefer all documents sharing the same set of latent topics.

On the other hand, the interaction with the system has been specialized for the analysts by defining a set of Dashboards designed to enable richer queries avoiding the complexity of their definition. Each Dashboard is designed to determine the *relatedness* between two homogeneous or heterogeneous texts based on a *Structured Semantic Text Similarity* function, discussed in [7]. This function measures the similarity between documents that are modeled according to record-like structures, so that each document is represented as a collection of different textual fields written in natural language. As an example, let us consider a user interested in searching useful Project Ideas or Grants for a given Organization. This activity is modeled as the task of retrieving all Project documents that

[2] FP7-SME-2010-1, Research for the benefit of specific groups, GA n. 262491

are considered related, i.e. compatible in some way, to the given Organization description. In the scenario targeted by PROGRESS-IT, the structural decomposition has been carried out as each section of different documents has different scope and importance. As an example, the decomposition may be useful to focus only on the competences of an organization, so ignoring its generic description, to estimate the relatedness with the foreseen activities in the project.

The resulting system is based on Solr[3], an open source enterprise search platform from the Apache Lucene project. Progress-IT inherits the Solr advantages, it is highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying, automated failover and recovery, centralized configuration and more. In this paper we will evaluate the Advanced Retrieval functionalities, i.e. the Dashboards, while a broader manual validation is nearing completion.

In the rest of the paper Section 2 presents the Semantic Text Similarity among unstructured texts as well as Structured Semantic Text Similarity between documents. Section 3 presents the PROGRESS-IT platform architecture. Finally, Section 4 presents the first evaluations.

2 Structured Similarity for Advanced Semantic Search

Semantic Textual Similarity (STS) measures the degree of semantic equivalence between two phrases or texts. An effective method to compute similarity between sentences or semi-structured material may have many applications in Natural Language Processing [12] and related areas such as Information Retrieval, improving the effectiveness of semantic search engines [14]. The definition of effective similarity functions presents an interesting research topic and it has been considered in several international evaluation campaigns, as in [1, 2]. The Structured Similarity and Advanced Semantic Search functionalities are discussed in this Section.

2.1 Distributional Models for Semantic Text Similarity among unstructured texts

Among existing text similarity functions, the *Bag of Word* (*BoW*) similarity function emphasizes pure lexical information, expressed as the word overlap between texts. Such representation is very common in Information Retrieval, since [16], where documents are represented as vectors whose dimensions correspond to different words. Many weighting schemas can be applied; in the later evaluations, each dimension represents a boolean indicator of the presence or not of a word in a text. The similarity function between two texts is the cosine similarity between vector pairs.

In order to generalize the lexical information of texts, the *Latent Semantic* (*LS*) similarity function is also applied. Such generalization is needed to reduce

[3] `http://lucene.apache.org/solr/`

data sparseness that usually compromises recall, as documents may not contain one or more query terms. Different approaches to acquire word meaning through the Distributional Analysis of word in large-scale corpora have been defined, as discussed in [15, 18, 17]. In this work, lexical generalization is obtained by a co-occurrence Word Space built accordingly to the methodology described in [6] and [15]. A word-by-context matrix M is obtained through the corpus analysis and the *Latent Semantic Analysis* [11] technique is applied to it. The matrix M is decomposed through Singular Value Decomposition (SVD) [10] into the product of three new matrices: U, S, and V so that S is diagonal and $M = USV^T$. M is then approximated by $M_k = U_k S_k V_k^T$, where only the first k columns of U and V are used, corresponding to the first k greatest singular values. This approximation supplies a way to project a generic word w_i into the k-dimensional space using $W = U_k S_k^{1/2}$, where each row corresponds to the representation vector $\boldsymbol{w_i}$. The original statistical information about M is captured by the new k-dimensional space, which preserves the global structure while removing low-variant dimensions, i.e. distribution noise. Traditional Information Retrieval approaches apply the SVD decomposition to a term-by-document matrix. As discussed in [5] these spaces well capture topical relations between words. In order to capture more fine-grained relations, we preferred a word-by-context space, able to better capture quasi-synonymic relations, as discussed in [15]. Given two words w_1 and w_2, the similarity function σ is estimated as the cosine similarity between the corresponding projections $\boldsymbol{w_1}, \boldsymbol{w_2}$ in the space, i.e $\sigma(w_1, w_2) = \frac{\boldsymbol{w_1} \cdot \boldsymbol{w_2}}{\|\boldsymbol{w_1}\| \|\boldsymbol{w_2}\|}$. The result is that each word can be projected in the reduced Word Space and an entire text, e.g. a sentence, can be represented by applying an *additive linear combination*, in line with [13]. Finally, the resulting kernel function is still modeled as the cosine similarity between vector pairs, in line with [4].

The Word Space is also beneficial to improve the ranking quality and enable *Query Suggestion* and *Expansion*. Given a query term, it is geometrically represented into the space and the most related words are automatically retrieved by considering the nearest vectors in the space. These new terms can be proposed to the user or automatically selected to extend the query terms. Obviously, in the automatic expansion phase new terms should be penalized into the rank in order to prefer explicit user terms. Finally, if the query terms are considered as a pseudo-document, the additive linear combination can be also used to estimate the similarity with the retrieved documents within the Word Space. It allows to assign higher rank to all documents containing different words, still related with the query. The similarity between the query vector and each document is here used to re-weight the numerical score determining a retrieval system rank[4].

2.2 Comparing documents with Structured STS

Structured Semantic Text Similarity insists on records, i.e. a sequence of typed textual fields, rather than on a individual text. We model the similarity between

[4] In particular, the score produced by the Distributional models has been combined through the multiplicative operation with the solr score.

documents in term of similarity between semi-structured data. In fact, not all the document sections have the same importance for a reader. As an example, a paper title or abstract usually synthesizes the whole content, thus containing the most representative information of the entire paper. In order to assign a different importance to different sections, we decomposed each document type considered in the project. As an example, patents are decomposed in specific fields containing the *title*, *abstract*, *claims* and *full-description*; papers are decomposed in *title*, *abstract* and *full-text*; projects in *title*, *project activities* (e.g. the work-package list) and *project objectives*; organization descriptions in *description*, *organization competences* and *organization key interests*. This approach allows to weight and combine the contribution of different linguistic evidences from each field through independent similarity functions.

When considering document pairs sharing the same types, a **homogeneous similarity** function is applied to consider corresponding fields, i.e. the LS similarity between the two abstracts or the two description in the paper. This is a simpler case with respect to a **heterogeneous similarity** function that, instead, considers heterogeneous documents and different fields. As an example, when considering a patent and a paper each field of the paper is compared with each field of the patent. However, the similarity between the paper abstract with the patent claims may be more important. It means that selected fields provide different evidences to the overall similarity. Each similarity type corresponds to a specific function, that selects the required information features to satisfy the user need. Obviously not all field combinations are needed: specific weighting schemes are applied to avoid the introduction of noise.

Moreover, not all morpho-syntactic informations are extracted as features from some fields. Filters are applied to focus on specific syntactic categories or Named Entities (NEs) classes: they are textual mentions to specific real-world categories, such as of PERSONS (PER), LOCATIONS (LOC) or DATES. They are detected in a field and made available as feature to the corresponding kernel: this introduces a bias on typed measures and emphasizes specific semantic aspects (e.g. places LOC or persons PER, in *location* or *author* measures, respectively). For example, in the sentence "*The chemist R.S. Hudson began manufacturing soap in the back of his small shop in West Bomich in 1837*", when POS tag filters are applied, only verbs (V), nouns (N) or adjectives (J) can be selected as features. This allows to focus on specific actions, e.g. the verb "*manufacture*", entities, e.g. nouns "*soap*" and "*shop*", or some properties, e.g. the adjective "*small*". When Named Entity categories are used, a mention to a person like "*R.S. Hudson*" or to a location, e.g. "*West Bomich*', or date, e.g. "*1837*", can be useful to model finer grain information. In the corpus collected in PROGRESS-IT, few documents contain mention to standard Named Entity classes and the real contribution of this specific information is still an open issue.

The combination of different fields from the document decomposition as well as the proliferation of functions that take into account specific morpho-syntactic information require a proper combination of fields. This can be learned from labeled data examples by learning *regression functions* to determine the proper

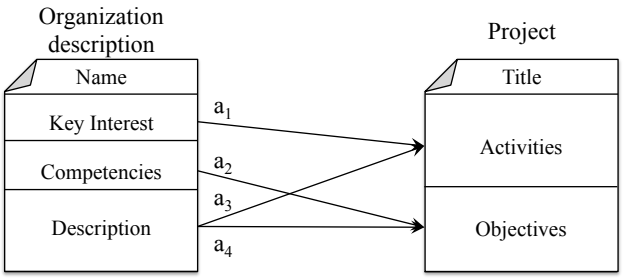

Fig. 1. Example of Structured Semantic Text Similarity

weighting, as discussed in many systems reported in [1, 2]. When the examples required for training the regressor are not available, a manual weighting can still be applied. Moreover, it allows the final user to personalize his own similarity function, even through simple graphical controls, e.g. sliders. An example of the Structured similarity estimated between two document types is shown in Figure 1. The similarity between an organization description and a project is shown. In the final score not all the fields are considered, but only a specific field subset is selected and weighted according to $a_1, ..., a_n$. In Figure 1, the similarity between the organization *Key interests* and project *Activities* is estimated and weighted by a_1, while the organization *Description* and the project *Objectives* similarity is weighted by a_4. Notice that some fields, such as the organization title, are not considered, i.e. they provide no contribution in the resulting score.

3 The PROGRESS-IT platform architecture

In this Section the architecture of the PROGRESS-IT platform is presented. It is summarized in Figure 2 and it can be divided in three main parts: the *Data Gathering* modules collect and process all the targeted documents; the *Persistence* modules store all this information into Solr cores; the *Retrieval* modules realize all the functionalities to retrieve documents, improve the rank quality and automatically build advanced queries; finally the *Human Interface* modules realize the interactions with the final users.

3.1 The Data Gathering modules

The Data Gathering modules collect targeted documents from heterogeneous sources, process texts, extract semantic representation and index documents. Linguistic analysis and distributional algebraic methods acquire semantic information and domain-specific lexicons for an accurate document search and ranking. The chain can be decomposed in the following modules:

– The **Import Handler** loads and pre-processes the requisite documents, in order to acquire a representation that is readable by the following modules.

Documents are extracted from database, such as Innovation Place within Ciaotech s.r.l.[5], specialized sites, such as EPO[6] for patent or the web, for example to retrieve scientific papers.

- The **Reveal Natural Language Toolkit** (**RevNLT**) is provided by Reveal s.r.l.[7] and implements techniques for natural language processing (NLP) used to achieve a morphosyntactic analysis of texts contained within documents. Examples of this analysis are the segmentation of the documents into sentences or the identification of the main classes providing grammatical characterization of the words that make up the sentences (e.g. nouns, verbs or adjectives). In the overall architecture, RevNLT represents a module providing linguistic information useful to build artificial representation for indexing/retrieval modules. During PROGRESS-IT the NLP processor has been customized for documents from three targeted domains, i.e. Aerospace, ICT and Security: specific lexicons and a domain specific terminology have been acquired; moreover all models have been updated for best results in the selected domains.
- The **Index System** is a first interface to serialize documents into the Solr cores. It selects all useful linguistic information provided by the RevNLT, decomposes each document according to its type (e.g. a patent or paper) and collects required information for each field to enable the advanced dashboards, as discussed in Section 2.2. All material is pre-processed in order to be easily handled by the Persistence layer (explained below).
- The **Advanced Retrieval and discovery** module implements the distributional algebraic methods needed to acquire the geometrical representation of word from the document collection. The statistical analysis of the entire corpus (made now of more than 200k documents) is performed. These informations are extracted in separated indexes and serialized in the Solr cores.

3.2 The Persistence modules

These modules implement all methods required to serialize extracted information and indexes into the Solr cores. We defined the required Solr schema for each document in order to reflect the required decomposition. Moreover, several functionalities, such as the weight of query terms and the first ranking function, are implemented within this low-level interface to Solr. They allow to efficiently retrieve the subset of indexed documents needed by each query in order to drastically improve performances.

3.3 The Retrieval modules

The Retrieval modules realize all functionalities to expand users queries, retrieve documents, improve the rank quality and automatically build advanced queries. The chain can be decomposed in the following modules:

[5] `www.innovationplace.eu`
[6] `www.epo.org`
[7] `www.revealsrl.it`

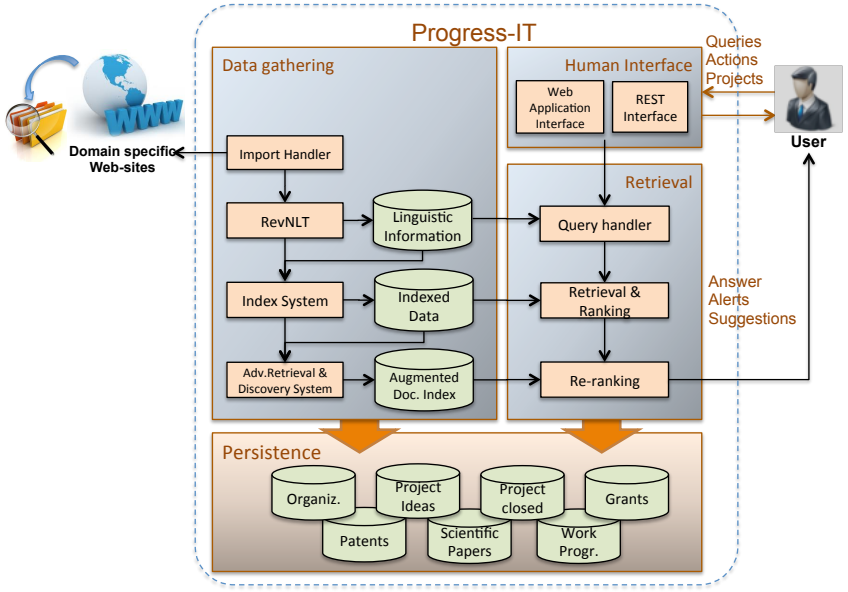

Fig. 2. The Progress-IT architecture.

- The **Query Handler** module intercepts and pre-processes the user query. As discussed in Section 2, the distributional representation of words enables the Query Suggestion. Moreover, it implements the Query Expansion step, thus extending the terms used to query the system, in order to increase the number of retrieved documents from the Solr cores and improving the system recall. The corresponding precision drop is reduced through the Semantic Re-ranking.
- The **Retrieval and ranking** module translates the user query for the retrieval system and stores documents. In PROGRESS-IT it acts as an interface for the different Solr cores to weight the importance of different query terms.
- The **Re-ranking** module enables the Advanced Retrieval functionalities. It allows to geometrically represent the query terms to re-rank the document list proposed by Solr. Moreover it implements the structured similarity between specific document types, to enable the Advanced Dashboards.

3.4 Human Interface

Finally, the human interface implements all possible interactions between the user and the system. Two interaction modalities are considered: a **Web Application** server enables the interaction via Web with the users; a **Rest-based Architecture** provides a set of SOA interfaces to other systems.

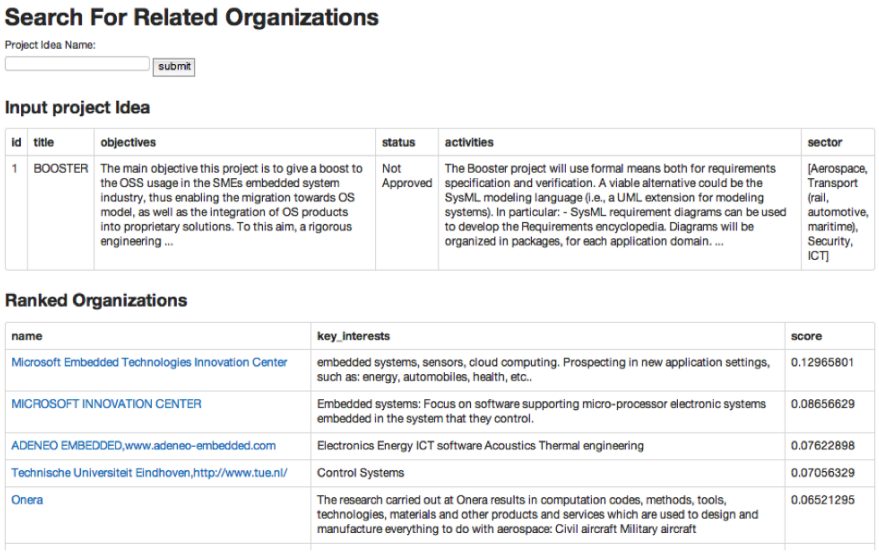

Fig. 3. *Project ideas* to *Organization descriptions* dashboard

4 Experimental evaluation

At the moment of writing the PROGRESS-IT platform has collected and indexed more than 350k document of many different types divided in:

- 150 Project Ideas
- 100k documents reporting Closed Project
- 400 Organization Descriptions
- 400 Grants
- 200k Patents
- 50k Papers
- Work Programmes from the European Community

We are now evaluating and validating the platform considering all possible user interactions. In this paper we will numerically evaluate some of the implemented Dashboards. In particular we implemented several Dashboards as shown in the following list, where the first document type represents the query while the second reflects the list of expected results:

- Project ideas to Organization descriptions
- Organization descriptions to Project Ideas
- Organization description To Grants
- Organization descriptions to Scientific Papers

In Figure 3 the first version of the Dashboard visualization is shown. In particular the *Project ideas* to *Organization descriptions* dashboard is considered.

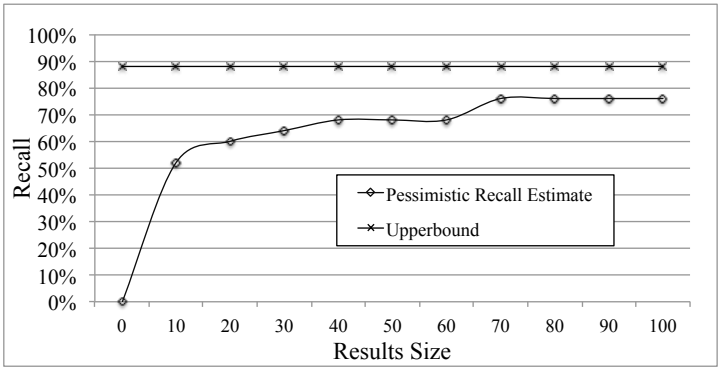

Fig. 4. Pessimistic Recall Estimate

The user can select the input project, shown in detail on the top of Figure 3, and the system provides the list of the related organizations, sorted according to the score shown in the last column of the table on the bottom of the figure. While the manual validation is not finished yet, a first automatic evaluation will be reported. We considered the "Organizations to Closed Projects" dashboard to measure the potential contribution that PROGRESS-IT would have given if used in the past years.

We simulated a consultant asking the system for possible projects for 25 random organizations. Nowadays these projects are funded but we think that the platform would have been useful if returning a project that the organization participated to. Obviously, it is a sort of pessimistic evaluation as one organization may have not participated to the project even being a good candidate. In Figure 4 this first "pessimistic" evaluation is reported. It reflects the percentage of organizations receiving at least a "good" project after a given number of projects. It means that 5 years ago, given the organization description, a consultant using PROGRESS-IT would have read only 10 project descriptions in order to find a good candidate project with more than 50% of recall. Some of the organizations did not participate to any projects from the indexed one, so determining an upper-bound of 90%. At the moment of writing a manual validation of the other dashboards is nearing completion.

5 Conclusion

The system targeted in the PROGRESS-IT project is a first application of some of the results of the European INSEARCH EU project[8], presented in [8] and it focuses on the design and development of a useful Search Platform for the Small and Medium Sized Enterprises (SMEs).

[8] FP7-SME-2010-1, Research for the benefit of specific groups, GA n. 262491

The PROGRESS-IT platform allows to access information derived from heterogeneous sources, through Standard or Advanced Information Retrieval techniques. The quality of the *Keyword Search* has been improved by applying Distributional models of Lexical Semantics, while an advanced query set has been defined and several *Dashboards* have been designed to enable richer queries, avoiding the complexity of their definition.

Acknowledgment

This work has been partially supported by the Regione Lazio under the project PROGRESS-IT (FILAS-CR-2011-1089).

References

1. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A.: Semeval-2012 task 6: A pilot on semantic textual similarity. In: *SEM 2012. pp. 385–393. Montréal, Canada (7-8 June 2012), http://www.aclweb.org/anthology/S12-1051
2. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W.: *sem 2013 shared task: Semantic textual similarity. In: Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity. pp. 32–43. Association for Computational Linguistics, Atlanta, Georgia, USA (June 2013)
3. Cocchi, L., Bohm, K.: Deliverable 2.2: Analysis of functional and market information. TECH-IT-EASY (2009)
4. Cristianini, N., Shawe-Taylor, J., Lodhi, H.: Latent semantic kernels. J. Intell. Inf. Syst. 18(2-3), 127–152 (2002)
5. Croce, D., Filice, S., Basili, R.: Distributional models and lexical semantics in convolution kernels. In: Gelbukh, A.F. (ed.) CICLing (1). Lecture Notes in Computer Science, vol. 7181, pp. 336–348. Springer (2012)
6. Croce, D., Previtali, D.: Manifold learning for the semi-supervised induction of framenet predicates: an empirical investigation. In: GEMS 2010. pp. 7–16. Association for Computational Linguistics, Stroudsburg, PA, USA (2010)
7. Croce, D., Storch, V., Basili, R.: Unitor-core_typed: Combining text similarity and semantic filters through sv regression. In: *SEM 2013. pp. 59–65. Association for Computational Linguistics, Atlanta, Georgia, USA (June 2013)
8. DeCao, D., Storch, V., Croce, D., Basili, R.: Insearch: A platform for enterprise semantic search. In: Basili, R., Sebastiani, F., Semeraro, G. (eds.) IIR. CEUR Workshop Proceedings, vol. 964, pp. 104–115. CEUR-WS.org (2013)
9. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. JASIS 41(6), 391–407 (1990)
10. Golub, G., Kahan, W.: Calculating the singular values and pseudo-inverse of a matrix. Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis 2(2), pp. 205–224 (1965)
11. Landauer, T., Dumais, S.: A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychological Review 104 (1997)
12. Mihalcea, R., Corley, C., Strapparava, C.: Corpus-based and knowledge-based measures of text semantic similarity. In: In AAAI06 (2006)

13. Mitchell, J., Lapata, M.: Composition in distributional models of semantics. Cognitive Science 34(8), 1388–1429 (2010)
14. Sahami, M., Heilman, T.D.: A web-based kernel function for measuring the similarity of short text snippets. In: Proceedings of the 15th international conference on World Wide Web. pp. 377–386. WWW '06, ACM, New York, NY, USA (2006), http://doi.acm.org/10.1145/1135777.1135834
15. Sahlgren, M.: The Word-Space Model. Ph.D. thesis, Stockholm University (2006)
16. Salton, G., Wong, A., Yang, C.: A vector space model for automatic indexing. Communications of the ACM 18, 613–620 (1975)
17. Turian, J., Ratinov, L.A., Bengio, Y.: Word Representations: A Simple and General Method for Semi-Supervised Learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. pp. 384–394. Association for Computational Linguistics, Uppsala, Sweden (Jul 2010)
18. Turney, P.D., Pantel, P.: From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research 37, 141–188 (2010)